CLAIMS:

What is claimed is:

1. A system for accessing and executing instruction sequences in a physical memory from a virtual memory in a processor-based system, comprising:

a memory for storing instruction sequences by which the processor-based system is processed, the memory including a physical memory and a virtual memory; and

a processor for executing the stored instruction sequences; and

wherein the stored instruction sequences include process steps to cause the processor to: (a) map a plurality of predetermined instruction sequences from the physical memory to the virtual memory; (b) determine an offset to one of said plurality of predetermined instruction sequences in the virtual memory; (c) receive an instruction to execute the one of said plurality of predetermined instruction sequences; (d) transfer control to the one of said plurality of predetermined instruction sequences; and (e) process the one of said plurality of predetermined instruction sequences from the virtual memory.

2. The system of Claim 1, wherein in step (c), the instruction is made from an application program.

3. The system of Claim 1, wherein in step (c), the instruction is made from a class driver.

4. The system of Claim 1, wherein step (a) comprises the steps of:

(a.1) mapping a plurality of BIOS instruction sequences from the physical memory to the virtual memory, said BIOS instruction sequences including a BIOS service directory; and

(a.2) mapping BIOS data from the physical memory to the virtual memory.

5. The system of Claim 4, wherein step (b) comprises the steps of:

(b.1) determining a starting virtual address of the BIOS service directory; and

(b.2) determining a starting virtual address of one of the plurality of BIOS instruction sequences by reference to the BIOS service directory.

6. The system of Claim 5, wherein step (d) comprises the steps of:

(d.1) creating a register stack in a memory location;

(d.2) identifying a location of the starting virtual address of one of the plurality of BIOS instruction sequences in the register stack; and

(d.3) transferring control to the one of the plurality of BIOS instruction sequences.

7. The system of Claim 6, wherein in step (d.1), the memory location is a buffer located in a dynamic random access memory (DRAM).

8. The system of Claim 6, wherein in step (d.1), the memory location is a buffer located in a main memory.

9. The system of Claim 6, wherein step (e) comprises the steps of:

(e.1) determining if the starting virtual address is within a range of addresses mapped from the physical memory to the virtual memory; and

(e.2) if so, executing the one of the plurality of BIOS instruction sequences from the virtual memory, otherwise indicating that the starting virtual address is not within the range of addresses mapped from the physical memory to the virtual memory.

10. A method for accessing and executing instruction sequences in physical memory from virtual memory in a processor-based system, comprising the steps of:

(a) mapping a plurality of predetermined instruction sequences from the physical memory to the virtual memory;

(b) determining an offset to one of said plurality of predetermined instruction sequences in the virtual memory;

(c) receiving an instruction to execute the one of said plurality of predetermined instruction sequences;

(d) transferring control to the one of said plurality of predetermined instruction sequences; and

(e) processing the one of said plurality of predetermined instruction sequences from the virtual memory.

11. The method of Claim 10, wherein in step (c), the instruction is made from an application program.

12. The method of Claim 10, wherein in step (c), the instruction is made from a class driver.

13. The method of Claim 10, wherein step (a) comprises the steps of:

(a.1) mapping a plurality of BIOS instruction sequences from the physical memory to the virtual memory, said BIOS instruction sequences including a BIOS service directory; and

(a.2) mapping BIOS data from the physical memory to the virtual memory.

14. The method of Claim 13, wherein step (b) comprises the steps of:

(b.1) determining a starting virtual address of the BIOS service directory; and

(b.2) determining a starting virtual address of one of the plurality of BIOS instruction sequences by reference to the BIOS service directory.

15. The method of Claim 14, wherein step (d) comprises the steps of:

(d.1) creating a register stack in a memory location;

(d.2) identifying a location of the starting virtual address of one of the plurality of BIOS instruction sequences in the register stack; and

(d.3) transferring control to the one of the plurality of BIOS instruction sequences.

16. The method of Claim 15, wherein in step (d.1), the memory location is a buffer located in a dynamic random access memory (DRAM).

17. The method of Claim 15, wherein in step (d.1), the memory location is a buffer located in a main memory.

18. The method of Claim 15, wherein step (e) comprises the steps of:

(e.1) determining if the starting virtual address is within a range of addresses mapped from the physical memory to the virtual memory; and

(e.2) if so, executing the one of the plurality of BIOS instruction sequences from the virtual memory, otherwise indicating that the starting virtual address is not within the range of addresses mapped from the physical memory to the virtual memory.

19. Computer-executable process steps for accessing and executing instruction sequences in physical memory from virtual memory in a processor-based system, the process steps including:

(a) mapping a plurality of predetermined instruction sequences from the physical memory to the virtual memory;

(b) determining an offset to one of said plurality of predetermined instruction sequences in the virtual memory;

(c) receiving an instruction to execute the one of said plurality of predetermined instruction sequences;

(d) transferring control to the one of said plurality of predetermined instruction sequences; and

(e) processing the one of said plurality of predetermined instruction sequences from the virtual memory.

20. Computer-executable process steps of Claim 19, wherein in step (c), the instruction is made from an application program.

21. Computer-executable process steps of Claim 19, wherein step (a) comprises the steps of:

(a.1) mapping a plurality of BIOS instruction sequences from the physical memory to the virtual memory, said BIOS instruction sequences including a BIOS service directory; and

(a.2) mapping BIOS data from the physical memory to the virtual memory.

22. Computer-executable process steps of Claim 21, wherein step (b) comprises the steps of:

(b.1) determining a starting virtual address of the BIOS service directory; and

(b.2) determining a starting virtual address of one of the plurality of BIOS instruction sequences by reference to the BIOS service directory.

23. Computer-executable process steps of Claim 22, wherein step (d) comprises the steps of:

(d.1) creating a register stack in a memory location;

(d.2) identifying a location of the starting virtual address of one of the plurality of BIOS instruction sequences in the register stack; and

(d.3) transferring control to the one of the plurality of BIOS instruction sequences.

24. Computer-executable process steps of Claim 23, wherein in step (d.1), the memory location is a buffer located in a dynamic random access memory (DRAM).

25. Computer-executable process steps in Claim 23, wherein in step (d.1), the memory location is a buffer located in a main memory.

26. Computer-executable process steps of Claim 23, wherein step (e) comprises the steps of:

(e.1) determining if the starting virtual address is within a range of addresses mapped from the physical memory to the virtual memory; and

(e.2) if so, executing the one of the plurality of BIOS instruction sequences from the virtual memory, otherwise that the starting virtual address is not within the range of addresses mapped from the physical memory to the virtual memory.

27. A system for accessing instruction sequences in a physical memory from a virtual memory in a processor-based system, comprising:

a memory for storing instruction sequences by which the processor-based system is processed, the memory including a physical memory and a virtual memory; and

a processor for executing the stored instruction sequences; and

wherein the stored instruction sequences include process steps to cause the processor to: (a) map a plurality of predetermined instruction sequences from the physical memory to the virtual memory; (b) determine an offset to one of said plurality of predetermined instruction sequences in the virtual memory; (c) receive an instruction to execute the one of said plurality of predetermined instruction sequences; (d) transfer control to the one of said plurality of predetermined instruction sequences; and (e) process the one of said plurality of predetermined instruction sequences from the virtual memory.

28. The system of Claim 27, wherein step (a) comprises the steps of:

(a.1) mapping a plurality of BIOS instruction sequences from the physical memory to the virtual memory, said BIOS instruction sequences including a plurality of BIOS read only memory (ROM) instruction sequences and a BIOS service directory; and

(a.2) mapping BIOS data from the physical memory to the virtual memory.

29. The system of Claim 28, wherein step (b) comprises the steps of:

(b.1) determining a starting virtual address of the BIOS service directory; and

(b.2) determining a starting virtual address of one of the plurality of BIOS instruction sequences by reference to the BIOS service directory.

30. The system of Claim 29, wherein step (d) comprises the steps of:

(d.1) creating a register stack in a memory location and;

(d.2) identifying a location of the starting virtual address of one of the plurality of BIOS ROM instruction sequences in the register stack.

31. The system of Claim 30, wherein step (e) comprises the steps of:

(e.1) determining if the starting virtual address is within a range of addresses mapped from the physical memory to the virtual memory; and

(e.2) if so, reading the one of the plurality of BIOS ROM instruction sequences from the virtual memory, otherwise indicating that the starting virtual address is not within the range of addresses mapped from the physical memory to the virtual memory.

32. A method for accessing instruction sequences in physical memory from virtual memory in a processor-based system, comprising the steps of:

(a) mapping a plurality of predetermined instruction sequences from the physical memory to the virtual memory;

(b) determining an offset to one of said plurality of predetermined instruction sequences in the virtual memory;

(c) receiving an instruction to execute the one of said plurality of predetermined instruction sequences;

(d) transferring control to the one of said plurality of predetermined instruction sequences; and

(e) processing the one of said plurality of predetermined instruction sequences from the virtual memory.

33. The method of Claim 32, wherein step (a) comprises the steps of:

(a.1) mapping a plurality of BIOS instruction sequences from the physical memory to the virtual memory, said BIOS instruction sequences including a plurality of BIOS read only memory (ROM) instruction sequences and a BIOS service directory; and

(a.2) mapping BIOS data from the physical memory to the virtual memory.

34. The method of Claim 33, wherein step (b) comprises the steps of:

(b.1) determining a starting virtual address of the BIOS service directory; and

(b.2) determining a starting virtual address of one of the plurality of BIOS instruction sequences by reference to the BIOS service directory.

35. The method of Claim 34, wherein step (d) comprises the steps of:

(d.1) creating a register stack in a memory location; and

(d.2) identifying a location of the starting virtual address of one of the plurality of BIOS ROM instruction sequences in the register stack.

36. The method of Claim 35, wherein step (e) comprises the steps of:

(e.1) determining if the starting virtual address is within a range of addresses mapped from the physical memory to the virtual memory; and

(e.2) if so, reading the one of the plurality of BIOS ROM instruction sequences from the virtual memory, otherwise indicating that the starting virtual address is not within the range of addresses mapped from the physical memory to the virtual memory.

37. A system to securely utilize Basic Input and Output System (BIOS) services, comprising:

an access driver to generate a service request to utilize BIOS services, the service request including a service request signature created using a private key in a cryptographic key pair; and

an interface to verify the service request signature using a public key in the cryptographic key pair to ensure the integrity of the service request.

38. The system of Claim 37, wherein:

the access driver generates a session request to establish a session with the interface; and

the session request includes a session request signature created using a private key in a cryptographic key pair.

39. The system of Claim 37, wherein:

the access driver generates a session request to end the session with the interface; and

the session request includes a session request signature created using a private key in a cryptographic key pair.

40. The system of Claim 37, wherein:

the interface generates an authority certificate and sends the authority certificate to the access driver after receiving a session request; and

the access driver uses information included in the authority certificate to generate subsequent session requests.

41. The system of Claim 40, wherein the authority certificate includes a new public key.

42. The system of Claim 40, wherein the authority certificate includes a new private key.

43. The system of Claim 40, wherein the authority certificate includes a certificate signature.

44. The system of Claim 37, wherein:

the interface generates an authority certificate and sends the authority certificate to the access driver after receiving the service request; and

the access driver uses information in the authority certificate to generate subsequent service requests.

45. A method to securely invoke Basic Input and Output System (BIOS) services, comprising:

creating a service request to invoke BIOS services;

signing the service request with a service request signature generated using a private key in a cryptographic key pair; and

verifying the service request signature using a public key in the cryptographic key pair to ensure the integrity of the service request.

46. The method of Claim 45, further comprising:

creating an authority certificate that includes a new private key and a new public key after processing the service request;

signing a subsequent service request with a service request signature generated using the new private key; and

verifying the service request signature of the subsequent service request using the new public key.

47. The method of Claim 45, further comprising:

performing a BIOS service indicated by a service operation code included in the service request.

48. The method of Claim 45, further comprising:

creating a session request to establish a session with a ROM Application Program Interface (RAPI);

signing the session request with a session request signature generated using a private key in a cryptographic key pair; and

verifying the session request signature using a public key in the cryptographic key pair to ensure the integrity of the session request.

49. The method of Claim 48, further comprising:

creating an authority certificate that includes a new private key and a new public key after processing the session request;

signing a subsequent session request with a session request signature generated using the new private key; and

verifying the session request signature of the subsequent session request using the new public key.

50. The method of Claim 45, further comprising:

creating a session request to end a session with a ROM Application Program Interface (RAPI);

signing the session request with a session request signature generated using a private key in a cryptographic key pair; and

verifying the session request signature using a public key in the cryptographic key pair to ensure the integrity of the session request.

51. A computer program embodied on a computer-readable medium to securely utilize Basic Input and Output System (BIOS) services, comprising:

an access driver to generate a service request to utilize BIOS services, the service request including a service request signature created using a private key in a cryptographic key pair; and

an interface to verify the service request signature using a public key in the cryptographic key pair to ensure the integrity of the service request.

52. A computer data signal embodied in a data stream, comprising:

an access driver to generate a service request to utilize BIOS services, the service request including a service request signature created using a private key in a cryptographic key pair; and

an interface to verify the service request signature using a public key in the cryptographic key pair to ensure the integrity of the service request.